

IN THE CLAIMS:

- 1 1. (CURRENTLY AMENDED): A method for fast reboot of a computer having an at-
2 tached disk array and an internal random access memory (RAM) comprising the steps of:
3 | retaining a copy of an operating system kernel at ~~on~~ a reserved storage location of
4 the RAM;
5 performing predetermined reboot operations with a boot mechanism; and
6 | reloading ~~the~~ an operative operating system at a location in the RAM based upon
7 the copy of the operating system kernel retained at the reserved storage location, after
8 the step of performing the predetermined reboot operations.
- 1 2. (ORIGINAL): The method as set forth in claim 1 wherein the boot mechanism is
2 adapted to perform predetermined full reboot steps based upon a full reboot instruction
3 from the operating system, and wherein the predetermined reboot operations omit the
4 predetermined full reboot steps when the operating system is reloaded from the reserved
5 storage location based upon a warm reboot instruction so as to perform a less-than-full
6 reboot.
- 1 3. (CURRENTLY AMENDED): The method as set forth in claim 2 wherein the prede-
2 termined full reboot steps include loading into the memory a copy of the operating sys-
3 tem kernel from the attached disk array ~~on-disk data from which the operating system~~
4 ~~kernel is reloaded~~, and the predetermined full reboot steps further include at least one of
5 (a) fully clearing of the RAM, including the reserved storage location, and (b) fully test-
6 ing the RAM.

1 4. (ORIGINAL): The method as set forth in claim 3 wherein the predetermined full re-
2 boot steps further include testing at least one of an LCD display chip and a Serial In-
3 put/Output (SIO) chip.

1 5. (CURRENTLY AMENDED): The method as set forth in claim 2 wherein the copy of
2 ~~in-memory data-~~ the operating system kernel comprises a compressed image of the oper-
3 ating system kernel adapted to be uncompressed and extracted to form the operative op-
4 erating system kernel at the location in the RAM.

1 6. (CURRENTLY AMENDED): The method as set forth in claim 5 wherein the step of
2 reloading includes loading the compressed image of the operating system kernel from the
3 reserved storage location to a space within the RAM and thereafter uncompressing and
4 extracting the operative operating system kernel into the RAM from the compressed im-
5 age of the operating system loaded into the RAM.

1 7. (ORIGINAL): The method as set forth in claim 6 further comprising, after the step of
2 uncompressing and extracting, freeing-for-overwrite the space within the RAM into
3 which the compressed image of the operating system kernel is loaded.

1 8. (CURRENTLY AMENDED): The method as set forth in claim 6 wherein one of the
2 predetermined full reboot steps comprises copying the compressed image of the operating
3 system kernel into the reserved storage space from the compressed image of the operating
4 system kenrel in the RAM.

1 9. (CURRENTLY AMENDED): The method as set forth in claim 2 wherein one of the
2 predetermined reboot steps comprises loading the ~~in-memory copy of the data-~~ operating

3 | system kernel at the reserved storage location based upon a ~~disk-stored~~ copy of the oper-
4 | ating sytem kernel ~~data~~ on the attached disk array ~~of disks~~.

1 10. (ORIGINAL): The method as set forth in claim 2 further comprising providing a
2 warm reboot instruction in response to a condition that enables the less-than-full reboot
3 of the computer.

1 11. (ORIGINAL): The method as set forth in claim 10 wherein the condition includes at
2 least one of a user-generated warm reboot command and a predetermined software panic
3 condition that can be repaired by the less-than-full reboot of the computer.

1 12. (ORIGINAL): The method as set forth in claim 10 wherein the step of providing the
2 warm reboot instruction includes setting a flag in the boot mechanism from a full reboot
3 state to a warm reboot state.

1 13. (ORIGINAL): The method as set forth in claim 10 further comprising reverting to
2 the full reboot if the copy of the data is corrupted.

1 14. (CURRENTLY AMENDED): A system for performing a fast reboot of a computer
2 | having an attached disk array ~~of attached disks~~ and an internal memory comprising:

3 | a boot mechanism that carries out full reboot operations on the computer, the boot
4 | mechanism including a flag adapted to indicate performance by the boot mechanism of
5 | either (a) the full reboot operations or (b) warm reboot operations wherein at least one of
6 | the full reboot operations is skipped, including a loading of data from the ~~array of disks~~
7 | disk array that generates the operating system kernel at a predetermined kernel location in
8 | the memory.

1 15. (PREVIOUSLY PRESENTED): The system as set forth in claim 14 wherein the full
2 reboot operations that are skipped include a full test of the memory, a zeroing of the
3 memory, and a shutdown of a processor of the computer .

1 16. (PREVIOUSLY PRESENTED): The system as set forth in claim 14 wherein the
2 memory includes a reserved storage space that stores a copy of data from which the oper-
3 ating system is loaded into the predetermined location in the memory during the warm
4 reboot operations, the boot mechanism being adapted to retain uncleared the reserved
5 storage space during the warm reboot operations, and being adapted to clear the reserved
6 storage space during the full reboot operations.

1 17. (ORIGINAL): The system as set forth in claim 16 wherein the copy of the data
2 stored in the reserved storage space is a compressed image of the operating system kernel
3 and the boot mechanism is adapted to load the compressed image of the operating system
4 kernel from the reserved storage space into a space in the memory outside the reserved
5 storage space, and further comprising a boot loader that subsequently uncompresses and
6 extracts the compressed image of the operating system kernel at the space in the memory
7 outside the reserved storage space to generate the operating system kernel at the prede-
8 termined kernel location in the memory.

1 18. (CURRENTLY AMENDED): A computer-readable medium including program in-
2 structions executing on a computer for fast reboot of a computer having an attached disk
3 array and an internal random access memory (RAM), the program instructions perform-
4 ing the steps of:

5 retaining a copy of an operating system kernel at in-memory data, from which a
6 ~~operating system kernel is reloaded~~, on a reserved storage location of the RAM;

7 performing predetermined reboot operations with a boot mechanism; and
8 reloading ~~the~~ an operative operating system kernel at a location in the RAM based
9 upon the copy of the operating system kernel retained at the reserved storage location af-
10 ter the step of performing the predetermined reboot operations.

1 19. (ORIGINAL): The computer-readable medium as set forth in claim 18 wherein the
2 boot mechanism is adapted to perform predetermined full reboot steps based upon a full
3 reboot instruction from the storage system, and wherein the predetermined reboot opera-
4 tions omit the predetermined full reboot steps when the operating system is reloaded from
5 the reserved storage location based upon a warm reboot instruction so as to perform a
6 less-than-full reboot.

1 20. (CURRENTLY AMENDED): The computer-readable medium as set forth in claim
2 19 wherein the predetermined full reboot steps include loading into the memory a copy of
3 ~~on-disk data from which the operating system kernel is reloaded~~ from the attached disk
4 array, and the predetermined full reboot steps further include at least one of (a) fully
5 clearing of the RAM, including the reserved storage location, and (b) fully testing the
6 RAM.

1 21. (ORIGINAL): The computer-readable medium as set forth in claim 20 wherein the
2 predetermined full reboot steps further include testing at least one of an LCD display chip
3 and a Serial Input/Output (SIO) chip.

1 22. (CURRENTLY AMENDED): A method for carrying out a warm reboot on a com-
2 puter having a random access memory (RAM) and an attached disk array ~~of attached~~
3 ~~disks~~ that skips predetermined full reboot steps comprising the steps of:

4 directing a boot mechanism that carries out the full reboot steps to undergo a
5 warm reboot;

6 retaining an uncleared reserved storage space in the RAM for storing a copy of an
7 ~~so as to store in memory data from which the operating system kernel is generated~~; and

8 after predetermined warm reboot steps are performed, generating an operative the
9 operating system kernel in a portion of the RAM outside the reserved storage space from
10 the copy of the operating system kernel in memory data in a portion of the RAM outside
11 the reserved storage space.

1 23. (CURRENTLY AMENDED): The method as set forth in claim 22 wherein the step of
2 generating ~~the operating system kernel~~ includes uncompressing a compressed image of
3 the operating system kernel.

1 24. (CURRENTLY AMENDED): The method as set forth in claim 23 wherein the step of
2 generating further includes copying the ~~in-memory data to form the~~ compressed image of
3 the operating system kernel ~~at to~~ a location within the portion of the RAM outside the
4 reserved storage location.

1 25. (CURRENTLY AMENDED): The method as set forth in claim 23 wherein the step of
2 generating includes performing an error check on the ~~data copy of the operating system~~
3 kernel and, if the ~~in-memory data~~ copy of the operating system kernel is corrupted, re-
4 verting to the full reboot steps.

1 26. (CURRENTLY AMENDED): The method as set forth in claim 23 wherein the full
2 reboot steps include (a) clearing the reserved storage space, (b) loading, from the disk
3 array, a copy of on-disk data from which the operating system kernel is generated and (c)

4 | copying the ~~on-disk data copy of the operating system kernel~~ to the reserved storage
5 | space, ~~to thereby store the in-memory data.~~

1 27. (ORIGINAL): The method as set forth in claim 26 wherein the full reboot steps fur-
2 ther comprise at least one of (a) fully clearing the memory, and (b) performing a full test
3 of the memory.

1 28. (ORIGINAL): A method for fast reboot of a computer having the random access
2 memory comprising the steps of:

3 performing predetermined reboot steps with a boot mechanism; and

4 reloading an operating system kernel extracted from a stored operating system
5 kernel image, the operating system kernel image being stored in a reserved location of the
6 random access memory prior to the step of performing the predetermined reboot steps,
7 the reserved location remaining uncleared after the step of performing the predetermined
8 reboot steps.

1 29. (ORIGINAL): A method for rebooting a computer comprising the steps of:

2 in response to a predetermined reboot command, selectively reloading an operat-
3 ing system kernel to a memory of the computer from either a first compressed kernel im-
4 age of the operating system or a second compressed kernel image of the operating sys-
5 tem, each of the first compressed kernel image and the second compressed kernel image
6 being located on different non-removable storage media, wherein an access speed for the
7 first compressed kernel image is quicker than an access speed for the second compressed
8 kernel image.

1 30. (CURRENTLY AMENDED): The method as set forth in claim 29 further compris-
2 ing storing the first compressed kernel image ~~is on~~ a reserved space of the memory that
3 remains uncleared during a warm reboot process having predetermined warm reboot
4 steps and storing the second compressed kernel on a disk operatively interconnected to
5 the computer.

1 31. (ORIGINAL): The method as set forth in claim 30 wherein the step of storing the
2 first compressed kernel image includes loading a copy of the second compressed kernel
3 image onto the reserved space during a full reboot process that includes both full reboot
4 steps and the predetermined warm reboot steps.

1 32. (ORIGINAL): The method as set forth in claim 31 wherein the full reboot steps in-
2 clude clearing the memory, including the reserved storage space and testing each of an
3 LCD display chip and a serial input/output (SIO) chip.

1 33. (CURRENTLY AMENDED): An operating system for a computer having an oper-
2 ating system kernel stored in a random access memory of the computer comprising:
3 a warm reboot instruction, responsive to a predetermined reboot condition, that
4 sets an indicator in a boot mechanism of the computer to perform a warm reboot process
5 that includes predetermined boot steps that are fewer than the boot steps performed by the
6 boot mechanism in response to a full reboot instruction; and
7 a compressed kernel image located at a reserved storage space in the random ac-
8 cess memory, the compressed kernel image being adapted to be accessed to reload the
9 compressed kernel image into the random access memory during a warm reboot process.

1 34. (CURRENTLY AMENDED): The operating system as set forth in claim 33 wherein
2 the boot steps of the full reboot process include mechanisms for fully clearing the random
3 access memory and reloading a compressed kernel image of the operating system from a
4 disk into the random access memory.

1 35. (CURRENTLY AMENDED): The operating system as set forth in claim 34 wherein
2 the compressed kernel image is located at a reserved storage space in the random access
3 memory that is remote from the area controlled by the operating system and that remains
4 uncleared during the warm reboot process.

1 36. (ORIGINAL): The operating system as set forth in claim 33 wherein the warm re-
2 boot instruction is based upon a condition in the computer that requires only the boot
3 steps of the warm reboot process and that allows the boot steps of the full reboot process
4 to be skipped.

1 37. (ORIGINAL): The operating system as set forth in claim 33 wherein the indicator
2 includes a plurality of warm reboot levels that enable each of the boot steps of the full
3 reboot process to be selectively performed or skipped.

1 38. (PREVIOUSLY PRESENTED) The method as in claim 1, wherein the attached disk
2 array is one or more disks.

1 39. (CURRENTLY AMENDED) The system as in claim 14, wherein the ~~array of at-~~
2 tached disk array ~~disks~~ is one or more disks.

1 40. (PREVIOUSLY PRESENTED) The computer-readable media as in claim 18,
2 wherein the attached disk array is one or more disks.

1 41. (CURRENTLY AMENDED) The method as in claim 22, wherein the ~~array of at-~~
2 tached disk array ~~disks~~ is one or more disks.

1 42. (PREVIOUSLY PRESENTED) A method for fast reboot of a computer, the com-
2 puter having an operating system loaded from a first memory, the method comprising the
3 steps of:

4 saving a copy of an operating system in a second memory, the second memory
5 having faster access than the first memory;

6 performing predetermined reboot operations; and

7 reloading the operating system from the operating system copy in the second
8 memory.

1 43. (PREVIOUSLY PRESENTED) The method as in claim 42, further comprising the
2 step of: compressing the copy of the operating system in the second memory.

1 44. (PREVIOUSLY PRESENTED) The method as in claim 43, further comprising the
2 step of: uncompressing the copy of the operating system.

1 45. (PREVIOUSLY PRESENTED) The method as in claim 42, wherein the copy of the
2 operating system is located at a reserved storage space in the second memory that is re-
3 mote from an area controlled by the operating system and that remains uncleared during
4 the fast reboot process.

1 46. (PREVIOUSLY PRESENTED) The method as in claim 42, wherein the step of per-
2 forming predetermined reboot operations further comprises skipping at least one step se-
3 lected from the group consisting of: performing a full memory test, reloading an operat-
4 ing system from the first memory, clearing system memory, testing an LCD display chip,
5 and testing a serial input/output (SIO) chip.

1 47. (PREVIOUSLY PRESENTED) The method as in claim 42, further comprising: set-
2 ting a flag indicating the fast reboot.

1 48. (PREVIOUSLY PRESENTED) The method as in claim 42, wherein the second
2 memory is a volatile memory.

1 49. (PREVIOUSLY PRESENTED) The method as in claim 48, wherein the second
2 memory is a RAM.

1 50. (PREVIOUSLY PRESENTED) The method as in claim 42, wherein the second
2 memory is a non-volatile memory.

1 51. (PREVIOUSLY PRESENTED) The method as in claim 50, wherein the second
2 memory is an NVRAM.

1 52. (PREVIOUSLY PRESENTED) The method as in claim 50, wherein the second
2 memory is a flash memory.

1 53. (PREVIOUSLY PRESENTED) The method as in claim 42, wherein the first mem-
2 ory is a disk memory.

1 54. (PREVIOUSLY PRESENTED) A computer for performing a fast reboot, the com-
2 puter comprising:

3 a first memory;

4 an operating system loaded from the first memory;

5 a second memory to save a copy of the operating system, the second memory hav-
6 ing faster access than the first memory; and

7 a boot mechanism to perform predetermined reboot operations, including reload-
8 ing the operating system from the operating system copy in the second memory.

1 55. (PREVIOUSLY PRESENTED) The computer as in claim 54, wherein the operating
2 system is compressed in the second memory.

1 56. (PREVIOUSLY PRESENTED) The computer as in claim 55, wherein the operating
2 system is uncompressed by the boot mechanism.

1 57. (PREVIOUSLY PRESENTED) The computer as in claim 54, further comprising: a
2 reserved storage space in the second memory that is remote from an area controlled by
3 the operating system and that remains uncleared during the fast reboot process, the re-
4 served storage space to save the copy of the operating system.

1 58. (PREVIOUSLY PRESENTED) The computer as in claim 54, wherein the predeter-
2 mined reboot operations further comprise skipping at least one step selected from the

3 group consisting of: performing a full memory test, reloading an operating system from
4 the first memory, clearing system memory, testing an LCD display chip, and testing a
5 serial input/output (SIO) chip.

1 59. (PREVIOUSLY PRESENTED) The computer as in claim 54, further comprising: a
2 flag to indicate the fast reboot.

1 60. (PREVIOUSLY PRESENTED) The computer as in claim 54, further comprising: a
2 volatile memory as the second memory.

1 61. (PREVIOUSLY PRESENTED) The computer as in claim 60, further comprising: a
2 RAM as the second memory.

1 62. (PREVIOUSLY PRESENTED) The computer as in claim 54, further comprising: a
2 non-volatile memory as the second memory.

1 63. (PREVIOUSLY PRESENTED) The computer as in claim 62, further comprising: an
2 NVRAM as the second memory.

1 64. (PREVIOUSLY PRESENTED) The computer as in claim 62, further comprising: a
2 flash memory as the second memory.

1 65. (PREVIOUSLY PRESENTED) The computer as in claim 54, further comprising: a
2 disk memory as the first memory.

1 66. (PREVIOUSLY PRESENTED) A computer for performing a fast reboot, the com-
2 puter comprising:

3 a first memory;

4 an operating system loaded from the first memory;

5 means for saving a copy of an operating system in a second memory, the second
6 memory having faster access than the first memory;

7 means for performing predetermined reboot operations; and

8 means for reloading the operating system from the operating system copy in the
9 second memory.

1 67. (PREVIOUSLY PRESENTED) The computer as in claim 66, further comprising: a
2 volatile memory as the second memory.

1 68. (PREVIOUSLY PRESENTED) The computer as in claim 66, further comprising: a
2 non-volatile memory as the second memory.

1 69. (PREVIOUSLY PRESENTED) The computer as in claim 66, further comprising: a
2 disk memory as the first memory.

1 70. (PREVIOUSLY PRESENTED) A computer readable media, comprising: said com-
2 puter readable media containing instructions for execution in a processor on a computer
3 having an operating system loaded from a first memory, the instructions for the practice
4 of the method of,

5 saving a copy of an operating system in a second memory, the second memory
6 having faster access than the first memory;

7 performing predetermined reboot operations; and
8 reloading the operating system from the operating system copy in the second
9 memory.

1 71. (PREVIOUSLY PRESENTED) Electromagnetic signals propagating on a computer
2 network, comprising: said electromagnetic signals carrying instructions for execution in a
3 processor on a computer having an operating system loaded from a first memory, the in-
4 structions for the practice of the method of,
5 saving a copy of an operating system in a second memory, the second memory
6 having faster access than the first memory;
7 performing predetermined reboot operations; and
8 reloading the operating system from the operating system copy in the second
9 memory.